



Solo POST

Transcripción

[00:00] Hola a todos y todas, bienvenidos y bienvenidas una vez más a este curso de Servlets. En el video pasado vimos sobre las diferencias entre POST y GET, vimos que GET sirve para obtener información normalmente, y se envían parámetros a través de la URL, mientras que POST nosotros utilizamos un formulario y la idea de POST es enviar informaciones al servidor.

[00:24] Claro que con un formulario nosotros también podemos utilizar el método GET, pero bueno, si queremos hacer un POST es sí o sí a través de un formulario, entonces vamos a ver un ejemplo de Alura, vamos a ver que cuando nosotros buscamos alguna cosa como Java, vean que acá arriba esa parte de búsqueda es un formulario, y vean que acá arriba en la URL se envió parámetros, query es el nombre del parámetro y el valor es Java.

[00:57] Entonces si nosotros hacemos clic derecho, inspeccionar encima del formulario de búsqueda vean que tenemos ese form donde no aparece ningún método, si no aparece ningún método específicamente, entonces significa que el formulario utiliza el método GET, mientras que por ejemplo si yo cierro acá sesión, vemos que tenemos la parte del login, y cuando nosotros hacemos clic derecho inspeccionar vean que tenemos ese formulario para iniciar sesión.

[01:30] Voy a bajar un poco más para que se vea mejor y utilizamos el método POST. Entonces ese formulario realmente nosotros no queremos que acá arriba se vean el usuario y la contraseña de la persona, entonces utilizamos ese método POST. Ahora vamos a ver qué es lo que nosotros tenemos.

[01:50] Perdón, estaba testeando unas cosas ahí. Acá en nuestro NuevaEmpresaServlet lo que nosotros tenemos es un service. Ese service atiende tanto a GET como POST. O sea, si nosotros, por ejemplo, nosotros enviamos a nuestra nueva empresa, a nuestro Servlet, servlet nuevaEmpresa, si nosotros enviamos nombre = Alura acá, va a funcionar.

[02:18] Y si nosotros enviásemos también por POST, también funcionaría. Entonces, lo que nosotros queremos es nosotros no queremos enviar los datos por GET porque no tiene sentido que nosotros enviemos una información usando GET. ¿Podemos hacerlo? Sí, podemos, pero no es una buena práctica. Entonces para eso lo que nosotros vamos a hacer es impedir que reciba un GET nuestro Servlet NuevaEmpresa. ¿Cómo hacemos eso?

[02:45] Bueno, en este service, nosotros lo que vamos a tener que hacer es cambiar ese service por un doPost. Vean que no pasa nada, no nos está dando error, no nos está dando ningún error sino que esto está todo bien, se ha recargado ahí el servidor. Esto lo que hace es obligar a que lo que nosotros recibimos en el Servlet sea un POST y no sea un GET.

[03:12] Si nosotros quisiésemos, por ejemplo, que sea sí o sí un GET y no un POST, podemos poner colocar doGet, pero no es este caso. Nosotros lo que queremos es realmente recibir nuestros datos a través de un POST, entonces, vamos a dejarlo como doPost y vamos a actualizar el servidor.

[03:31] Una vez iniciado de nuevo el servidor, vean qué es lo que ocurre. Si yo intento enviar a nuestro Servlet ese nombre = Alura como un GET, vean lo que obtenemos. Nosotros obtenemos un estado http 405. Nosotros vimos en los primeros videos el Estado 404, que era cuando encontraba una página, y ahora aprendimos el 405, este 405 es: no permitimos ese método.

[03:59] Y en el mensaje vean que no se nos dice explícitamente el método http GET no está soportado por esta URL. De esa forma entonces hemos conseguido evitar que una persona nos envíe una empresa a través del método GET. Ahora,

si nosotros nos vamos al formulario `formNuevaEmpresa`, esto debería continuar funcionando, ya que nosotros enviamos, déjenme ver acá, a través de `formNuevaEmpresa`.

[04:37] A través de ese `formNuevaEmpresa`, nosotros estamos utilizando el método POST, entonces esto de acá debería funcionar, vamos a intentar con otra empresa Caelum. Vamos a dar un enviar y vean que fue ejecutado correctamente. Empresa Caelum registrada. Y vean que acá, por ejemplo dentro de los datos enviados recibimos un 200.

[05:01] Un 200 significa que está todo bien, que funcionó todo bien, tanto el envío como la recepción de los datos, el formato, todo está bien, entonces ahí tienen otro estado también más, el cual aprendimos ahora. Perfecto. Y con esto ahora ya hemos aprendido un poco más sobre este mundo de Servlets. Espero que hagan los ejercicios y nos vemos en el siguiente video.